Technical Report TR30, May, 2004

Technical Report: OSU-CISRC-05/04-TR30
Department of Computer Science and Engineering
The Ohio State University
Columbus, OH 43210-1277

Ftp site: **ftp.cse.ohio-state.edu**
Login: **anonymous**
Directory: **pub/tech-report/2004**
File in pdf format: **TR30.pdf**
Web site: **http://www.cse.ohio-state.edu/research/tech-report.html**

Model-based Sequential Organization in Cochannel Speech

Yang Shao and *DeLiang Wang*

Department of Computer Science and Engineering
& Center for Cognitive Science
The Ohio State University
Columbus, OH 43210-1277, USA
{shaoy, dwang}@cis.ohio-state.edu

Abstract— A human listener has the ability to follow a speaker's voice while others are speaking simultaneously; in particular, the listener can organize the time-frequency energy of the same speaker across time into a single stream. In this paper, we focus on sequential organization in cochannel speech, or mixtures of two voices. We extract minimally corrupted segments, or usable speech, in cochannel speech using a robust multipitch tracking algorithm. The extracted usable speech is shown to capture speaker characteristics and improves speaker identification performance across various target-to-interferer ratios. To utilize speaker characteristics for sequential organization, we extend the traditional speaker identification framework to cochannel speech and derive a joint objective for sequential grouping and speaker identification, leading to a problem of search for the optimum hypothesis. Subsequently we propose a hypothesis pruning algorithm based on speaker models in order to make the search computationally feasible. Evaluation results show that the proposed system approaches the ceiling speaker identification performance obtained with prior pitch information, and yields significant improvement over alternative approaches on sequential organization.

Index Terms—Model-based approach, speaker identification, auditory scene analysis, cochannel speech, sequential organization.

I. INTRODUCTION

Cochannel speech is a combination of speech utterances from two talkers, usually produced when two speech signals are transmitted over a single communication channel. Unlike conversations, talkers from different channels are not aware of each other in cochannel speech. Consequently, speech from both channels has large overlap, which presents a considerable challenge to automatic speaker and speech recognition. On the other hand, for a cochannel recording that has comparable energies from both talkers (e.g. target-to-interferer ratio, or TIR, is zero), human listeners can readily select and follow one speaker's voice. Even in worse scenarios such as a cocktail-party, listeners can select and follow the voice of a particular talker [10], [7], [4]. Bregman [4] describes this process of auditory perception as auditory scene analysis, which is composed of simultaneous organization and sequential organization. The former integrates concurrent sound components and the latter integrates components across time into the same perceptual stream. Most of the existing computational auditory scene analysis systems, e.g. [5], [11], address only simultaneous organization. It is well known that human listeners use speaker characteristics, such as pitch and vocal tract information to identify a speaker's voice [22] and such characteristics have been incorporated in models of automatic speaker recognition [1], [16], [19], [9], [20]. In this paper, we study how to sequentially organize time-frequency energy of the same speaker into a single stream in cochannel speech using speaker characteristics, particularly, based on speaker models.

Research has been carried out for decades to extract one of the speakers from cochannel speech by either enhancing target speech or suppressing interfering speech [18], [17]. Zissman and Seward [27] examined pitch continuity in cochannel speech and assigned pitch contours to a corresponding talker by polynomial contour fitting when pitch contours from two speakers cross. Their results suggest that a method based purely on pitch information is not sufficient. Morgan et al. [17] estimated the dominant pitch and then reconstructed the speech components of both stronger and weaker talker frame by frame using frequency-domain filtering according to the estimated pitch; speech signals are further enhanced by the formants estimated for the stronger talker. Afterwards, a speaker assignment algorithm using a maximum likelihood criterion is applied to group recovered signals into two speaker streams, one for the target and the other for the interferer. The assignment algorithm groups the individual frames by examining the pitch and spectral continuity for consecutive voiced frames, and comparing the spectral similarity of the onset frame of a voiced segment with recently assigned frames using the divergence measure proposed by Carlson and Clement [6], which is the symmetrized Kullback-Leibler divergence [13]. Because of the short-term processing, the spectral comparison is biased towards the comparison of phonetic information contained in a frame instead of speaker characteristics. Therefore, to capture speaker characteristics, it is desirable to base comparison on speaker homogeneous segments, which consists of a number of time frames dominated by one speaker.

In automatic speaker recognition, as pointed out in [14], the intelligibility and quality of extracted speech are not important. What the system needs are portions of the speech that contain speaker characteristics unique to an individual speaker, classifiable and long enough for the system to make identification or verification decisions. These portions of speech, or segments, are defined as consecutive frames of speech that are minimally corrupted by interfering speech, and are thus called usable speech [14]. Previous studies [12], [14] find that voiced segments contain much information for speaker identification (SID), and have developed criteria such as frame-level TIR and spectral autocorrelation ratio to extract usable speech in cochannel mixtures. Results show that a significant amount of cochannel speech can be considered usable for SID. Frame TIRs are easily

calculated with premixing speech utterances, and usable speech extracted based on a TIR threshold produces frames in which energy from one speaker is much stronger than that of the other. Spectral autocorrelation ratio estimates the ratio between dominant peak and valley in the autocorrelation of spectrum in order to decide whether a frame is well structured (single-speaker speech) or unstructured (corrupted speech). Finally, the extracted usable segments are grouped using frame-level TIRs. It is a simple and effective method and shows a substantial improvement in SID performance. However, frame-level TIRs are hard to estimate from mixture speech.

Studies have been conducted on speaker detection and tracking in multi-speaker environments such as conversational speech and broadcast news. Various methods, supervised or unsupervised, have been explored. A typical method [8] is to use log-likelihood ratio scores, calculated from trained speaker GMMs and a universal background model, to partition a recording into homogeneous segments and then cluster the segments. However, such methods cannot be applied to cochannel speech because, as mentioned earlier, cochannel talkers strongly overlap, resulting in very short speaker-homogenous segments. In the case of 0 dB TIR, such segments typically last 30 *ms* to 300 *ms*, far shorter than the optimal segment length of around 2.5 *sec* and the typical minimum length of 1 *sec* for speaker clustering [8]. As pointed out in [14], a speaker recognizer's ability to identify talkers based on pooled frame-level scores is sharply reduced if available observation frames are limited in number, especially when the overall length is less than 500 *ms*. To verify this, we have explored segment clustering for sequential grouping ourselves; specifically, segments are iteratively clustered based on distance measures in the feature space, such as cepstral coefficients. The result is barely above the chance level.

In this paper, we propose to sequentially organize automatically extracted usable speech, i.e. speaker-homogeneous segments, into streams. Our method employs a robust multipitch tracking algorithm proposed recently [25] for extraction. We develop a computational objective for joint cochannel speaker identification and sequential grouping, or speaker assignment, of usable speech. Our formulation leads to a search problem to find an optimal hypothesis in the joint speaker and grouping space. Exhaustive search finds the optimal hypothesis though it is computationally infeasible. We propose a hypothesis pruning method, which iteratively removes hypotheses with low probabilities and thus reduces the search space and computation time greatly. We show that the pruning method achieves a performances level close to that of exhaustive search and ceiling performance with prior pitch information.

Our system is introduced in Section II. We describe how to extract usable speech using multipitch tracking in Section III. In Section IV, we develop the computational goal by extending the probabilistic framework of traditional speaker identification to cochannel speech and detail our method to achieve the objective. Evaluation results and comparisons are given in Section V. Section VI concludes the paper.

II. System Overview

In this section, we give an overview of the processing stages of our system. As shown in Figure 1, the proposed system consists of three stages. First, the multipitch tracking algorithm [25] is adapted and applied to cochannel speech and pitch contours for both speakers are produced. The algorithm filters the mixture signal into multiple frequency channels through an auditory filterbank; it then selects "clean" channels and peaks within each clean channel as pitch candidates at each time frame. Multiple pitch hypotheses are formed; the hypotheses are further integrated across the frequency channels. Afterwards, pitch contours are decoded as a sequence of most likely pitch

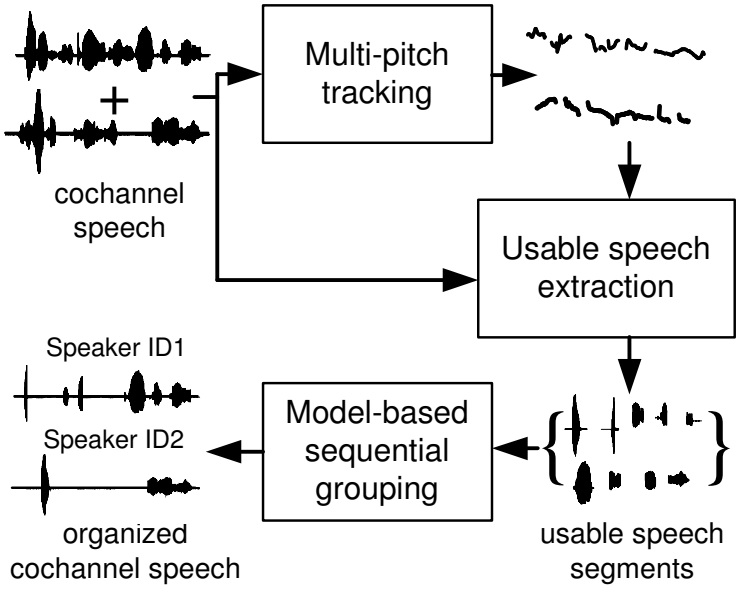

Figure 1. Schematic diagram of the proposed system. First, cochannel speech is passed through a multipitch tracking algorithm and pitch contours are obtained. Then usable speech segments are extracted based on the pitch information. Finally, a model-based sequential grouping algorithm organizes segments into two streams and corresponding speaker identities are also produced.

hypotheses using a HMM (hidden Markov Model) framework.

The second stage is to extract usable speech from a cochannel mixture based on the pitch information [23]. Due to the nature of human voice, a speech utterance contains voiced portions, unvoiced portions and silence. Therefore, there are some portions (segments) of cochannel speech that contain only one speaker's voiced part or one speaker's voiced part plus another speaker's unvoiced part, the latter usually having much lower energy. The voiced spectra of these frames are minimally corrupted, and can be used to derive speaker features for SID. So they form usable speech and are retained, while the portions with overlapping pitch contours as well as silent portions are removed, resulting in a set of usable speech segments.

For any two segments in the usable speech set, whether they are from the same speaker is unknown. In the third stage, our model-based sequential grouping algorithm groups the segments into two speaker streams by searching for the optimal hypothesis in the joint speaker and grouping space. Our formulation is extended from the traditional SID probabilistic framework. Exhaustive search in the space is computationally prohibitive with increasing number of segments. Thus we propose a hypothesis pruning algorithm to remove hypotheses of low likelihoods, which drastically reduces computation time while resulting in comparable performance with exhaustive search. As a byproduct, speaker identities are also determined.

III. Usable Speech Extraction via Multipitch Tracking

We employ and adapt a recent multipitch tracking algorithm proposed by Wu et al. [25] for usable speech extraction. We chose this algorithm because it is designed to track two overlapping pitch contours, which fits our needs, and produces very good results.

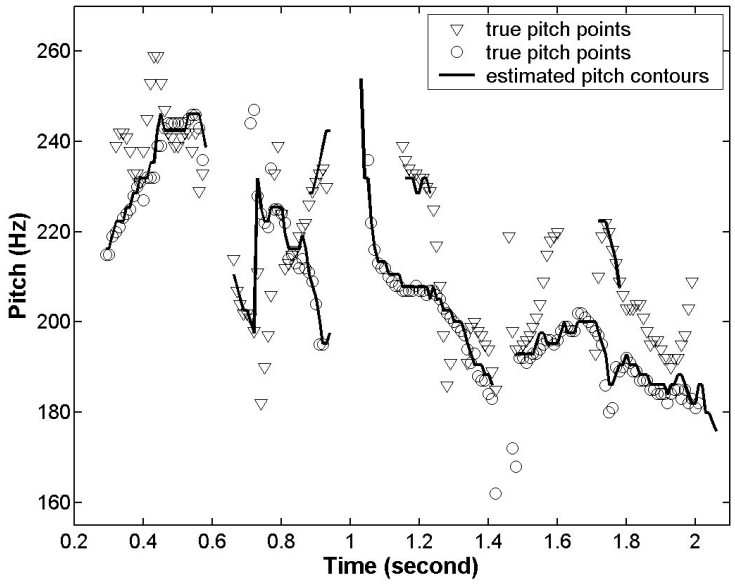

Figure 2. Estimated pitch contours from multipitch tracking compared with true pitch points. The solid lines represent the pitch contours obtained from a female-female co-channel mixture using the multipitch tracking algorithm. The triangles and circles represent the true pitch points obtained from the premixing utterances using Snack.

First, an input mixture is passed through a bank of 128 gammatone filters in order to obtain a cochlear spectrogram, or cochleogram, representation. The envelopes in high-frequency channels (center frequency greater than 800 Hz) are calculated and normalized correlograms (autocorrelations) are computed for each frequency channel. The peaks of the correlogram in a frequency channel indicate the periodicity of the signal, but some peaks are inconsistent with the pitch because of pitch dynamics and the fact that harmonics are unresolved in high frequency channels. Also, in noisy conditions the peaks in corrupted channels do not agree with the pitch. In order to minimize the effects introduced by these false peaks, corrupted channels are removed and the peaks are further selected in the retained clean channels.

A statistical model of pitch contours given the observed peaks is constructed as follows. A mixture of a Laplacian and a uniform distribution is employed to model the distribution of time-lag difference between the true pitch period and the closest peak in a selected channel. The distribution parameters are estimated from clean speech by maximum likelihood. Thus the probability of a frequency channel supporting a pitch hypothesis is formulated. An integration method is then used to produce the conditional probability of observing the selected peaks in all selected channels in a time frame given a hypothesized pitch period. A continuous HMM is used to model dynamic pitch contours. HMM states represent possible pitch states in every time frame and the transitions represent the probabilistic pitch dynamics, which models the pitch change in time and the jumps between zero-pitch, one-pitch and two-pitch spaces. The observation probability is the observed conditional probability described above.

Figure 2 shows an example of multipitch tracking. The cochannel speech is created by mixing two female utterances. The true pitch points are obtained using Snack [24] (an open source version of ESPS/waves+) from premixing utterances. The algorithm produces the pitch contours that fit well to the true pitch points, even though these two utterances have the same pitch ranges. It is

evident from the figure that in the mixture, there are portions that contain only one speaker's voiced speech, and portions that contain both speakers' voiced speech. There are also portions considered by the algorithm to contain one speaker's voiced speech but they actually contain both speakers' voiced speech. A typical reason for this mistake is that one speaker's voiced energy is much lower than that of the other. This kind of mistake, however, is rather benign as far as usable speech extraction is concerned.

Usable speech extraction means to determine what segments, i.e. sequences of frames, contain only a single speaker's information, thus usable for speaker identification. Pitch contours overlap from time to time due to the nature of cochannel speech. Pitch-overlapping segments are not usable for SID because the energies of both talkers are strong, leading to the corruption of single-speaker features used in SID. In such a frame, more precisely, the harmonics and formants from both talkers are added together in the power spectrum domain and ruin the second frequency analysis (discrete cosine transform) in the derivation of commonly used cepstral features. Speech enhancement methods such as spectral subtraction [3] are not effective here because human speech is highly nonstationary. Thus we remove pitch-overlapping segments from cochannel speech.

For the segments with only one speaker's voiced speech, the other speaker is either silent or producing unvoiced speech. In the former case, the power spectrum is intact; in the latter case, usually the energy of unvoiced speech is much lower than voiced speech and the voiced power spectrum is contaminated much less than in the voiced-voiced situation. Thus we consider these segments with single-pitch contours as usable speech. The remaining segments are considered unusable and removed. To ensure the homogeneity of a usable speech segment, if estimated pitch values of neighboring frames change abruptly, we consider that a speaker change occurs. Specifically, if this change is above 10 Hz, the segment is split into two shorter segments.

IV. Model-Based Sequential Organization

Maximum-likelihood classification is well established for speaker identification [19]. However, in order to recognize talkers in cochannel speech, the traditional probability framework for a single speaker needs to be extended to multiple speakers.

A. *Speaker Identification*

Given a set of reference speaker models $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_K\}$, the goal of SID is to find the speaker model that maximizes the posterior probability for an observation sequence, $O = \{o_1, o_2, \ldots, o_M\}$. Cepstral features, such as mel-frequency cepstral coefficients (MFCCs), are widely used as observations for speech signals. The speaker identification decision rule is

$$\hat{\lambda} = \arg\max_{\lambda \in \Lambda} P(\lambda \mid O). \tag{1}$$

Applying the Bayesian rule, we have

$$\hat{\lambda} = \arg\max_{\lambda \in \Lambda} \frac{P(O \mid \lambda) P(\lambda)}{P(O)}. \tag{2}$$

Typically, prior probabilities of speakers are assumed equal, and the probability of observing O is the same for all speakers in Λ. Using pre-trained speaker models and assuming independence between observations at different times, (2) can be rewritten as

$$\hat{\lambda} = \arg\max_{\lambda \in \Lambda} \sum_{m=1}^{M} \log p(o_m \mid \lambda) \tag{3}$$

after taking the log operation. Here m indexes observations. $p(o \mid \lambda)$ is the standard Gaussian mixture model (GMM) estimated from training speech of specific talkers using the EM algorithm [19].

B. Extension to Cochannel Speech

Cochannel speaker identification aims to find two speaker models that maximize the posterior probability for the observations. For a cochannel mixture, our usable speech extraction method extracts N speech segments, $X = \{S_1, S_2, \ldots, S_i, \ldots, S_N\}$, each of which is a segment of consecutive speech frames, $S_i = \{x\}$, with a single pitch contour. Given X, (1) can be modified as follows

$$\hat{\lambda}_{\mathrm{I}}, \hat{\lambda}_{\mathrm{II}} = \arg\max_{\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}} \in \Lambda} P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}} \mid X), \tag{4}$$

which is to find a pair of speaker models, $\hat{\lambda}_{\mathrm{I}}$ and $\hat{\lambda}_{\mathrm{II}}$, from the speaker set Λ that maximize the posterior probability given usable speech segments. As mentioned earlier, the single-pitch segments must be organized into two speaker streams because in cochannel speech one speaker can dominate in some portions and be dominated in other portions. For example, a possible segment assignment (grouping) may look like $\{S_1^0, S_2^1, \ldots, S_i^1, \ldots, S_N^0\}$, where superscripts, 0 and 1, do not represent the speaker identities but only denote that the segments marked with the same label are from the same speaker. Therefore, the joint computational objective of sequential grouping and SID may be stated as finding a pair of speaker models, $\hat{\lambda}_{\mathrm{I}}$ and $\hat{\lambda}_{\mathrm{II}}$, together with a segment assignment, $\hat{y}$, that jointly maximize the posterior probability:

$$\hat{\lambda}_{\mathrm{I}}, \hat{\lambda}_{\mathrm{II}}, \hat{y} = \operatorname*{argmax}_{\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}} \in \Lambda, y \in Y} P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}, y \mid X), \tag{5}$$

where Y is the assignment space, which includes all possible assignments (labelings) of the segments.

C. Derivation

The posterior probability in (5) can be rewritten as

$$P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}, y \mid X) = \frac{P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}, y, X)}{P(X)} = \frac{P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}, X^y)}{P(X)}, \tag{6}$$

in which X^y represents the labeled segments according to a specific segment assignment y. Eq. (6) can be further written as

$$P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}, y \mid X) = P(X^y \mid \lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}) \frac{P(\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}})}{P(X)}. \tag{7}$$

Assuming the independence of speaker models and using the same assumption from traditional speaker identification that prior probabilities of speaker models are the same, we insert equation (7) into (5) and remove the constant terms. The objective then becomes finding two speakers and an assignment that have the maximum probability of assigned usable speech segments given the corresponding speaker models as follows.

$$\hat{\lambda}_1, \hat{\lambda}_{11}, \hat{y} = \underset{\lambda_1, \lambda_{11} \in \Lambda, y \in Y}{\arg\max} P(X^y \mid \lambda_1, \lambda_{11}). \tag{8}$$

Note, the conditional probability is essentially the joint SID score of assigned segments. Assuming that any two segments, S_i and S_j, are independent of each other given the speaker models, the conditional probability in (8) can be written as

$$P(X^y \mid \lambda_1, \lambda_{11}) = \prod_{S_i \in X^0} P(S_i \mid \lambda_1) \prod_{S_j \in X^1} P(S_j \mid \lambda_{11}), \tag{9}$$

in which X^0 is the subset of usable speech segments labeled with 0; and X^1 the subset of segments labeled with 1. The probability of having a segment, S, from a pre-trained speaker model λ is the product of likelihoods of that speaker model generating each individual observation x of the segment, assuming the observations are independent of each other. In other words,

$$P(S \mid \lambda) = \prod_{x \in S} p(x \mid \lambda). \tag{10}$$

In the experiments of this paper, speakers are modeled as 16-mixture GMMs, which are tested to be sufficient for the data, and the observations or features used are MFCCs and their first-order dynamic coefficients [26].

Instead of the formulation in (6), the posterior probability can also be expanded as follows

$$P(\lambda_1, \lambda_{11}, y \mid X) = P(\lambda_1, \lambda_{11} \mid y, X) P(y \mid X). \tag{11}$$

The first term on the right side is the probability of the two speaker models given an assignment of usable speech segments, which is the joint SID score of the assigned segments for two speaker identities. The second term is the conditional probability of a specific assignment, y, given the usable segments, and this probability may be called the assignment model. The decomposition of the posterior in (11) is analogous to speech-recognition-based fragment grouping in [2].

D. Computational Method

The computational objective in (8) is to find two speakers and one assignment that yield the maximal probability using (9)-(10). Given the extracted usable speech segments and individual speaker models trained from clean speech, the maximization amounts to a search for the globally optimal hypothesis in the joint speaker and assignment space, Λ and Y.

The brute-force way to find the maximum is exhaustive search. For a cochannel mixture file, this involves calculating the probability of the assigned segments given a pair of speaker models, $P(X^y \mid \lambda_1, \lambda_{11})$, for every possible pair out of K speakers in Λ, and every assignment in Y of N segments. Each segment can take either label, 0 or 1. Therefore the search space grows with the number of speakers in a polynomial fashion, but it grows exponentially with the number of segments. Let the calculation of $P(X^y \mid \lambda_1, \lambda_{11})$ take a unit time, then total computation time is on the order of $O(\binom{K}{2} \cdot 2^N) = O(K^2 \cdot 2^N)$. Clearly, exhaustive search is computationally prohibitive with a large number of usable speech segments though it produces theoretically optimal results.

However, in the search space, some hypotheses have very low probabilities. Therefore, if these hypotheses could be identified and pruned from further consideration, the computation time could be greatly reduced. The results of exhaustive search indicate peaky distributions with each peak occupied by several assignment hypotheses in the search space. Thus, although only retaining the

best hypothesis is not optimal, keeping a small number of hypotheses appears sufficient. We propose an iterative hypothesis pruning method to keep only the two best hypotheses at each iteration that integrates all the assigned segments. Specifically, the algorithm is as follows.

Hypothesis Pruning Algorithm

Step 0. Order the segments in $X = \{S_1, S_2, \ldots, S_N\}$ in time.

Step 1. Label S_1 in X with 0 (assign it to X^0). This initial assignment is arbitrary.

Step 2. For S_2 in X, form two hypotheses: H_0, H_1, and create a label path for each of them. H_0 assumes that the current segment belongs to set X^0, and H_1 assumes that the current segment belongs to X^1. The label paths are

$$Path[2][H_0] = (0,0),\ Path[2][H_1] = (0,1).$$

$Path[n][\bullet]$ records assignment labels for the past n-1 segments and the hypothesized assignment of the current segment.

Step 3. For an unprocessed segment S_n, n>2, form H_0 and H_1. Then expand the label path for H_0 and H_1 as follows,

$$Path[n][H_0] = \left(Path[n-1]\left[\arg\max_{H \in \{H_0, H_1\}} L\big(Path[n-1][H], 0\big) \right], 0 \right)$$

$$Path[n][H_1] = \left(Path[n-1]\left[\arg\max_{H \in \{H_0, H_1\}} L\big(Path[n-1][H], 1\big) \right], 1 \right)$$

where the L function, as defined below, estimates the joint SID score by considering the best partial segment assignment from 1 to n.

$$L\big(Path[n-1][H], l\big) = \max_{\lambda_{\mathrm{I}}, \lambda_{\mathrm{II}} \in \Lambda} P(X^{(Path[n-1][H], l)} \mid \lambda_{\mathrm{I}}, \lambda_{\mathrm{II}}), \tag{12}$$

l = 0 or 1, refers to the hypothesized labeling for the current segment.

Step 4. Repeat Step 3 until the last segment S_N is processed. For S_N, compare the likelihood values returned by L for H_0 and H_1. The final winning hypothesis is the one with the higher likelihood. Obtain the corresponding two speaker identities that maximize (12) and the segment assignment for this hypothesis.

The L function in (12) is the same as (8) except that L only considers the partial segment assignment from S_1 to S_n. Figure 3 gives an illustration of this iterative algorithm. Since every usable segment could be produced by either of two speakers in the mixture, it is hypothesized as either H_0 or H_1 and labeled with 0 or 1 respectively (S_1 is initialized to hypothesis H_0). The two hypotheses bifurcate iteratively and our pruning algorithm always retains two best paths ending with different labels. The best label path up to the current hypothesis is recorded in *Path*. For H_0 and H_1 of a segment, we compare the partial SID scores, considering the label paths recorded with the preceding segment and the hypothesized assignment for the current hypothesis. The SID score is defined by the L function in (12). The better path is then chosen. The algorithm repeats until the last segment is considered.

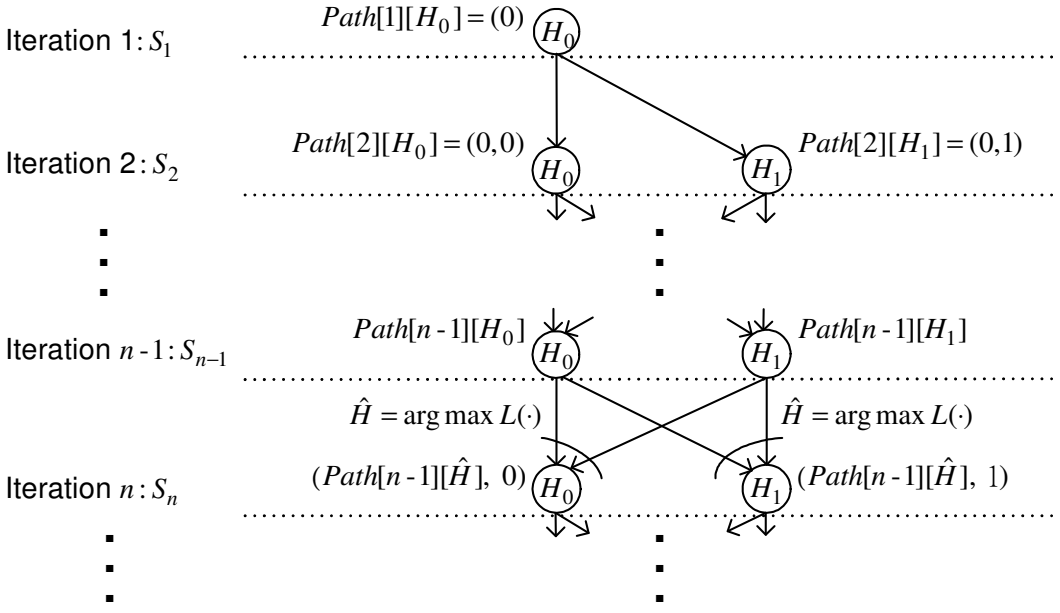

Figure 3. Illustration of the hypothesis pruning algorithm. The algorithm is executed segment by segment. Every segment is hypothesized to be either H_0 or H_1 and labeled with 0 or 1 respectively, except that S_1 is identified with hypothesis H_0. *Path* records the best label path. For either hypothesis of the segment to be considered, the better label path from the preceding iteration is chosen by comparing $L(\cdot)$ defined in Eq. (12), and its label path is copied to the current path. The algorithm repeats until the last segment.

Since at each iteration hypotheses are pruned according to the partial scores, this is a greedy algorithm and it is a form of beam search [21]. For each unlabeled segment, it retains two hypotheses, each of which calculates $P(X^y \mid \lambda_{\mathrm{I}}, \lambda_{\mathrm{II}})$ twice in the worst case, resulting in the polynomial time complexity on the order of $O\left(\begin{pmatrix} K \\ 2 \end{pmatrix} \cdot 2 \cdot N\right) = O(K^2 N)$.

V. Evaluation and Comparison

A. Data Preparation

As in Lovekin et al. [14], we employ the evaluation data from the TIMIT speech corpus. The speaker set consists of 38 speakers from the "DR1" dialect region, 14 of which are females and the rest are males. Each speaker has 10 utterance files, ranging from about 1.5 *sec* to 6.2 *sec* in length. For each speaker, 5 out of 10 files are used for training and the remaining 5 files are used to create cochannel mixtures for testing. For each speaker deemed as the target speaker, 1 out of 5 test files is randomly selected and mixed with randomly selected files of every other speaker, which are regarded as interfering utterances. For each pair the overall TIR of the speech mixture is calculated as the ratio of the target speech energy over the interfering speech energy,

$$\mathrm{TIR} = 10\log_{10}\left(\sum_t (s_T^2[t]) \Big/ \sum_t (s_I^2[t])\right),$$

in which s_T and s_I are the speech samples of target and interfering speakers in the time domain. Speech is scaled to create the mixtures at different TIRs: -20 dB, -10 dB, -5 dB, 0 dB, 5 dB, 10 dB and 20 dB. For example, 0 dB TIR means that the target speech overall energy is equal to that of the interfering speech. Thus, for each TIR, a total of 1406 cochannel mixture files are created for the testing purpose.

B. Usable Speech Evaluation

Our first experiment evaluates how the new method works for usable speech extraction. As the baseline performance, cochannel speech is fed into the speaker identification system without any processing. When one speaker's voice dominates the other as at very high or very low TIRs, it is likely that one speaker's model will give a higher score than the other so that the identification system classifies it as either the target speaker or the interfering speaker. We consider that the SID system make a correct decision if the cochannel speech is identified as either of the two underlying speakers. Then the cochannel speech is processed and usable speech segments are extracted as described previously. In this experiment, we assume that pitch information of individual speakers is known *a priori* and segments are grouped into speaker streams accordingly in order to test whether the extracted segments are useful for speaker identification. Specifically, a segment takes the label that is taken by the majority of the frames in it, which is determined by comparing the detected pitch with the *a priori* pitch. To mirror the decision for non-processing condition, the system is deemed to make a correct decision if either set of the usable segments is identified correctly.

Figure 4 shows the SID error rates with respect to TIR levels. Two observations can be made from the results. First, usable speech extraction substantially improves the identification performance. In the 0 dB TIR case, for example, the error rate is almost cut in half. Second, the improvements occur across all TIR levels. Improvement decreases at higher or lower TIRs because the dominating speaker is less interfered by the other speaker and the system tends to identify the dominating speaker correctly.

In some situations, one is more interested in the speaker from a specific channel (target speaker), so that the speech signal from the other speaker is considered interfering noise. Therefore, we perform the second experiment, which is the same as the first one except that the system is deemed to make a correct decision if the cochannel speech is identified as the target speaker, which is defined when the cochannel speech is created. Figure 5 presents results of the second experiment. Similar observations can be made from the results. First, under cochannel situations, usable speech extraction improves SID performances; the average improvement is about 14% in terms of absolute correct rate. Second, the improvements are consistent across all TIR levels. Improvement decreases at higher TIRs because the designated target speaker dominates the mixture. However, the target speaker is dominated by the interferer at lower TIRs, resulting in better performance after usable speech extraction.

C. Sequential Grouping Evaluation

Here we evaluate the performance of our model-based sequential organization approach. For this evaluation we only consider cochannel mixtures with overall TIR equal to 0 dB to simulate real cochannel situation.

As there are two types of output jointly produced from the system, we show the results separately in two tables. Table 1 shows the correct rate of speaker assignment by counting correctly assigned frames, which count those from the same speaker and marked with the same label. Table

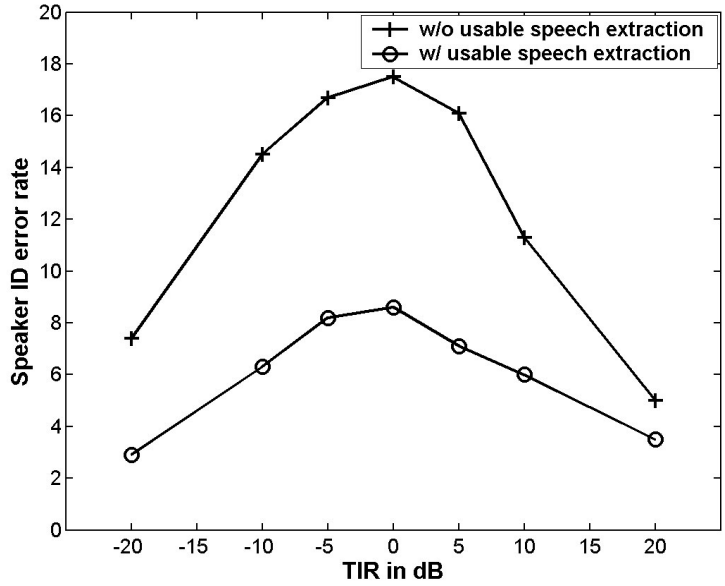

Figure 4. Speaker identification (SID) error rate before and after usable speech extraction. SID is considered correct when cochannel speech is identified as either of the two underlying speakers. Sequential grouping is done using *a priori* pitch information.

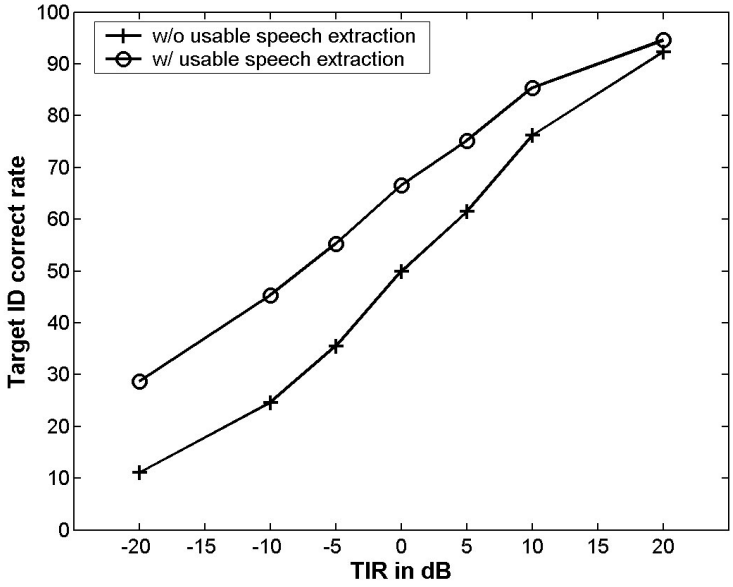

Figure 5. SID correct rate before and after usable speech extraction. SID is considered correct when the target speaker is identified from cochannel speech. Sequential grouping is done using *a priori* pitch information.

TABLE 1. CORRECT ASSIGNMENT RATE FOR DIFFERENT SEQUENTIAL GROUPING METHODS.

	Correct rate in percentage
W/o usable speech processing	50.0
Ideal assignment by prior pitch	94.1
Exhaustive search	77.4
Hypothesis pruning	76.2
Pitch dynamics	68.2
Spectral divergence	66.2

TABLE 2. COCHANNEL SPEAKER IDENTIFICATION CORRECT RATE.

	Speaker identification correct rate in percentage		
	Criterion I	Criterion II	Criterion III
W/o usable speech processing	N/A	50.0	82.5
Ideal assignment by prior pitch	43.3	72.0	93.7
Exhaustive search	40.2	70.4	93.9
Hypothesis pruning	37.5	68.8	93.0
Pitch dynamics	22.3	52.5	90.4

2 shows the SID performances with three different criteria because there are up to two speakers in a cochannel mixture. Criterion I records the percentage of mixtures where both speakers are correctly identified; this is the most stringent criteria. Like the evaluation in the preceding section, sometimes the speaker from a specified channel is of interest. Thus, criterion II displays target identification correct rate. Criterion III counts the files where either of the two speakers in the mixture is identified correctly.

In Table 1, under the test condition without usable speech processing, each frame can be marked either 0 or 1; so the baseline rate of correct grouping is 50%. The second row shows that ideal assignment by prior pitch achieves 94.1% correct rate. Note that ideal assignment is applied at the segment level and a segment takes the label of a majority of the frames in it, each decided by comparing the detected pitch with the prior pitch before mixing. The less-than-perfect result reflects that a single-pitch segment does not always contain frames from the same speaker, which is not surprising considering the nature of cochannel speech.

Exhaustive search achieves 77.4% correct assignment rate. It reflects the effectiveness of using speaker characteristics for sequential organization. From the derivation it is evident that exhaustive search places an upper limit on the performance of model-based sequential grouping. Our proposed hypothesis pruning method achieves 76.2% correct rate, approaching the upper limit set by exhaustive search.

Similar observations can be made from SID results in Table 2. The first row gives the baseline performance with unprocessed mixtures. As there is only the mixture, Criterion I does not apply. Ideal assignment produces the ceiling performance though it is not 100% correct because of

imperfect assignment and limited segment lengths. For the model-based approach, exhaustive search approaches the ceiling SID performance with ideal assignment, and the hypothesis pruning method performs almost as well as exhaustive search, while drastically cutting the overall computation time – from an average of 7 minutes per file to 0.7 seconds per file on a Pentium III workstation. Since the search is based on SID scores, the performance gap between the model-based method and ideal assignment is smaller than that of sequential grouping performance.

We have also explored variations of the hypothesis pruning algorithm. First, we have evaluated retaining just one hypothesis instead of two in step 3 of the algorithm. This essentially degrades the algorithm to local decision making, and it performs significantly worse than keeping two hypotheses. Instead of keeping the two best hypotheses ending with different labels, we have tried retaining two or three best hypotheses out of a total of four considering the hypotheses from previous iteration. The results are similar to those in the tables. The peaky hypothesis distribution in the search space is a main reason why our pruning method approaches the ceiling performance.

D. Comparison

We have shown the system's ability to extract usable speech and improve both cochannel SID and sequential organization performance. In this section, we compare with alternative sequential grouping methods, namely one that employs pitch dynamics and one based on spectral divergence.

One reasonable alternative is to utilize pitch information, particularly since pitch contours have already been obtained. Previous studies have demonstrated the importance of pitch contours for speaker recognition, e.g. [1]. We collect pitch differences between the end-point of a segment and the start-point of the following segment from the training data. Considering that the longer is the gap between two segments the less likely they belong to the same speaker, we multiply the difference by the time lag between them. The resulting product describes the pitch change dynamics between neighboring segments. A Gaussian-like peak can be observed centered on 0 in the histogram and maximum likelihood estimation is employed to obtain the statistics of the distribution, which is modeled as a mixture of Gaussian and uniform distribution [25]. When grouping, for each segment from S_1 to S_N, the pitch dynamics product is obtained and a local decision is made regarding whether the current segment comes from the same speaker as the previous segment by comparing the likelihoods of the dynamics feature given the distribution. After the assignment is done, a search for the two most probable speakers is applied. So it is obvious that the computational complexity is $O(K^2)$ for this method. The speaker assignment and SID results are presented in Tables 1 and 2. This method clearly performs worse than the pruning algorithm, but gives a significant improvement over the baseline case without usable speech processing.

We have also compared our algorithm with a spectrum-based method, specifically the speaker assignment algorithm of Morgan et al. [17] that also addresses sequential organization. Their system aims to enhance cochannel speech by separating two talkers and subsequently assigning separated speech components to two speaker streams. The assignment of intermittent voiced components, essentially the beginning frames of segments, is based on a frame-level spectral comparison with 50 recently assigned frames using the spectral divergence measure of Carlson and Clement [6]. Since our system considers a usable segment to belong to one speaker, we adapt and employ the algorithm to perform only speaker assignment; that is, segments are organized using their spectrum-based method. Specifically, the initial 50 frames of each speaker stream are *a priori* assigned, and then the subsequent segments are sequentially assigned according to their divergence measure. The assignment result is shown in Table 1. With 66.2% correct rate, the spectral method is comparable in performance to the pitch dynamics method, and it lags behind our proposed method.

VI. Concluding Remarks

Sequential organization groups sound components of the same source across time into the same stream. In this paper, we have proposed a model-based approach for sequential organization, to assign the extracted usable speech segments into speaker streams. Our usable speech extraction method produces segments useful for cochannel SID across various TIR conditions. We have shown that the proposed hypothesis pruning algorithm achieves SID performance close to the ceiling performance with prior pitch information or exhaustive search, and it performs significantly better than alternative approaches to speaker assignment.

It is worth noting that our sequential grouping algorithm can handle the situation where only one speaker is present in a cochannel mixture. Since segments may all take the same label after assignment, our algorithm can produce only one speaker identity. Also, the probabilistic framework proposed in here can be extended to situations with more than two speakers in a mixture.

The speech decoding model of Barker et al. [2] also addresses sequential integration, and their formulation is extended from the statistical framework of automatic speech recognition. Their model searches for the most likely word sequence and additionally determines the set of signal fragments that compose the speech signal, leaving the rest as the noise background. Our model is analogous to theirs in the emphasis of recognition-based organization. However, the domain of cochannel speaker recognition where our model is derived differs from their speech recognition domain, and as a result the computational methods used in the two models are very different. It is not clear, for example, how their model can address the cochannel situation where the interfering noise is also speech.

Acknowledgment

We thank M. Wu for his assistance in using the multipitch tracking algorithm. This research was supported in part by an AFOSR grant (FA9550-04-1-0117) and an NSF grant (IIS-0081058).

References

[1] B.S. Atal, "Automatic speaker recognition based on pitch contours," *J. Acoust. Soc. Am.*, vol. 52, pp. 1687-1697, 1972.

[2] J. Barker, M. Cooke, and D. Ellis, "Decoding speech in the presence of other sources," *Speech Comm.*, in press, 2004.

[3] M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of speech corrupted by acoustic noise," in *Proc. ICASSP*, pp. 113-120, 1979.

[4] A.S. Bregman, *Auditory scene analysis.* Cambridge MA: MIT Press, 1990.

[5] G.J. Brown and M. Cooke, "Computational auditory scene analysis," *Computer Speech and Language*, vol. 8, pp. 297-336, 1994.

[6] B.A. Carlson and M.A. Clements, "A computationally compact divergence measure for speech processing," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 1-6, 1991.

[7] E.C. Cherry, "Some experiments on the recognition of speech with one and with two ears," *J. Acoust. Soc. Am.*, vol. 25, pp. 975-979, 1953.

[8] R.B. Dunn, D.A. Reynolds, and T.F. Quatieri, "Approaches to speaker detection and tracking in conversational speech," *Digit. Sig. Process.*, vol. 10, pp. 93-112, 2000.

[9] S. Furui, *Digital speech processing, synthesis, and recognition.* New York: Marcel Dekker, 2001.

[10] H. Helmholtz, *On the sensation of tone.* Second English ed., New York: Dover Publishers, 1863.
[11] G. Hu and D.L. Wang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *IEEE Trans. Neural Net.*, in press, 2004.
[12] K.R. Krishnamachari, R.E. Yantorno, D.S. Benincasa, and S.J. Wenndt, "Spectral autocorrelation ratio as a usability measure of speech segments under cochannel conditions," in *Proc. ISPACS*, 2000.
[13] S. Kullback, *Information Theory and Statistics.* New York: Dover Publications, 1968.
[14] J.M. Lovekin, R.E. Yantorno, K.R. Krishnamachari, D.S. Benincasa, and S.J. Wenndt, "Developing usable speech criteria for speaker identification," in *Proc. ICASSP*, pp. 421-424, 2001.
[15] I. Masuda-Katsuse and H. Hawahara, "Dynamic sound stream formation based on continuity of spectral change," *Speech Comm.*, vol. 27, pp. 235-259, 1999.
[16] T. Matsui and S. Furui, "Text-independent speaker recognition using vocal tract and pitch information," in *Proc. ICASSP*, pp. 137-140, 1990.
[17] D.P. Morgan, E.B. George, L.T. Lee, and S.M. Kay, "Cochannel speaker separation by harmonic enhancement and suppression," *IEEE Trans. Speech Audio Process.*, vol. 5, pp. 407-424, 1997.
[18] T.F. Quatieri and R.G. Danisewicz, "An approach to co-channel talker interference suppression using a sinusoidal model for speech," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 38, pp. 56-69, 1990.
[19] D.A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Comm.*, vol. 17, pp. 91-108, 1995.
[20] D.A. Reynolds*, et al.*, "The SuperSID project: exploiting high-level information for high-accuracy speaker recognition," in *Proc. ICASSP*, pp. 784-787, 2003.
[21] S. Russell and P. Norvig, *Artificial intelligence: A modern approach.* 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2003.
[22] A. Schmidt-Nielsen and T.H. Crystal, "Human v.s. machine speaker identification with telephone speech," in *Proc. ICSLP*, 1998.
[23] Y. Shao and D.L. Wang, "Co-channel speaker identification using usable speech extraction based on multi-pitch tracking," in *Proc. ICASSP*, pp. 205-208, 2003.
[24] K. Sjolander and J. Beskow, "Wavesurfer - an open source speech tool," in *Proc. ICSLP*, 2000.
[25] M. Wu, D.L. Wang, and G.J. Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Trans. Speech Audio Process.*, vol. 11(3), pp. 299-241, 2003.
[26] S. Young, D. Kershaw, J. Odell, V. Valtchev, and P. Woodland, *The HTK Book* (*for HTK Version 3.0*). Microsoft Corporation, 2000.
[27] M.A. Zissman and D.C. Seward, "Two-talker pitch tracking for co-channel talker interference suppression," *Technical Report, Lincoln Laboratory, MIT*, 1992.